STAD68 Final Project

Binary Classifiers to Evaluate German Credit Risk

Clinton Ali

December 3, 2018

1 Data

1.1 Preprocess

The German Risk data (Fig 1) has 10 features, the features of the data are Age, Sex, Job, Housing, Saving accounts, Checking account, Credit amount, Duration and Purpose and the label of the data is called Risk. Since the features of the data are a mix of numerical and categorical values, it was necessary to transform the numerical values into a categorical one.

```
   Unnamed: 0  Age     Sex  Job Housing Saving accounts Checking account
0           0   67    male    2     own             NaN           little
1           1   22  female    2     own          little         moderate
2           2   49    male    1     own          little              NaN
3           3   45    male    2    free          little           little
4           4   53    male    2    free          little           little

   Credit amount  Duration             Purpose  Risk
0           1169         6            radio/TV  good
1           5951        48            radio/TV   bad
2           2096        12           education  good
3           7882        42 furniture/equipment  good
4           4870        24                 car   bad
```

Figure 1. Raw data

First of all, age was transformed into a categorical variable by discriminating if an applicant is in retirement age or not. Second, credit amount of the loan was transformed into a categorical data by discriminating the continuous values by using quantiles. All values up to the 1st quantile are considered low, values in the interquartile range is considered medium, and values greater than the 3rd quantile to be considered high. Third, duration of the loan was transformed from a discrete numericla value into a categorical value by grouping all loans less than 12 months to be a short term loan, and anything greater than or equal to 12 months to be a long term loan. After transforming all the variables into a categorical variable, it was then possible to use dummy variables to model the levels of the categories. By encoding the cateogrical variable using the one hot encoding (dummy variable) method, the features grew from 9 columns up to 31 columns to encode all the possible combinations of the categorical variables.

Subsequently, the data set was split into a 80% training and 20% testing set by simple random sample. This was employed to make sure that the test risk is an unbiased estimate of true risk. After the split it was found that the training set was imbalanced. In order to combat this, Synthetic Minority Over-sampling Technique (SMOTE) would be employed such that it would both oversample and undersample the training data in order to create a 50:50 balance. This would both exaggerate the minority and majority population of the training set and reduce estimation error due to bias.

2 Training

2.1 Methods

These five following classifiers are chosen to solve the task of binary classification: h_1: Logistic regression with a regularization parameter of 0.8 and the rest of the parameters are chosen as default.

h_2: Support vector machines with a polynomial kernel of degree 2, the rest of the parameters are chosen by default.

h_4: Random forest with 6 trees with maximum depth equal to the number of features after adding the dummy variables were chosen as parameters of this algorithm. The maximum depth was chosen because the data was transformed to a purely categorical one, and the maximum depth of the tree will enable all permutations of the categorical variables to work. The rest of the parameters are chosen as default.

h_5: K Nearest Neighbor with $k = 7$ and with default parameters for the rest.

h_6: Artifiicial Neural Network with a hidden layer, sigmoid activation function and Stochastic Gradient Descent to tune its weights as parameters. The rest of the hyper parameters are default values from the function itself.

2.2 Crossvalidation

After preprocessing the data and finally achieving a train and test split, it is then possible to employ cross validation on the training set. Stratified k-fold is chosen in order to preserve the 50:50 balance of the training data. In this data, four folds were chosen and all the algorithms were run through each permutation of the folds of the data once. The best classifier score out of the four folds will determine the best model of each algorithm which will then be used on the testing set.

3 Analysis

3.1 Empirical Risk

The empirical risk of each classifier were found to be as follows:
$L_S(h_1) = 0.275$
$L_S(h_2) = 0.345$
$L_S(h_3) = 0.320$
$L_S(h_4) = 0.345$
$L_S(h_5) = 0.300$

3.2 Hoeffding's Bound on Empirical Risk with a zero-one loss function

The Hoeffding's bounds were chosen because they offer a tighter bound compared to the usual methods. The confidence intervals were computed as follows:

$$P(|L_{S_{test}}(h_i) - L_S(h_i)|) \geq \sqrt{\frac{\log 2/\delta}{2m}} \text{ for } i \in 1,2,3,5,4,6$$

Where $m = 200$, or size of testing set.

Based on the one-at-time 95% and 99% confidence intervals (fig 2), logistic regression was the classifier with the tightest upper bound on the empirical risk. However, based on both te 95% and 99% simultaneous confidence intervals (fig 2), with equal weights given to each classifer, it is possible to conclude that there are no statistically significant difference between each classifier's empirical risk.

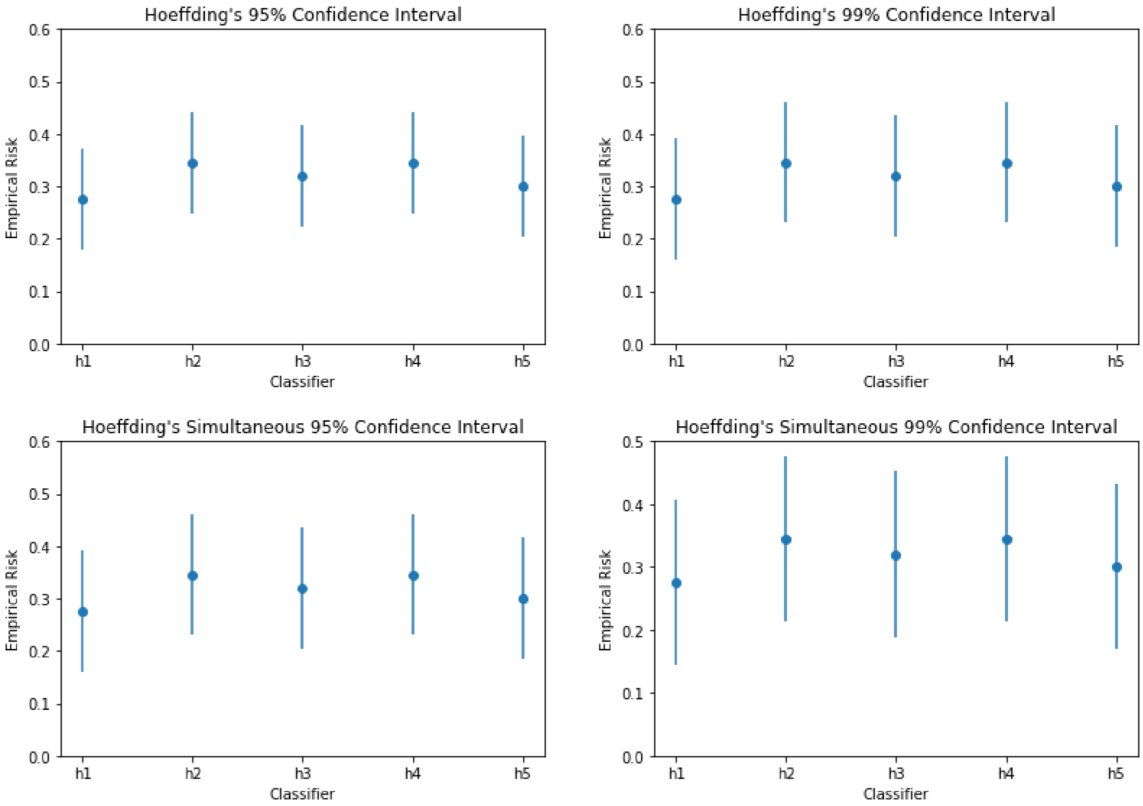

Figure 2. Hoeffding's bounds for Empirical Risk

3.3 Average Precision Score

Precision and recall curve measures the ratio of true positives over the sum of true and negative positives. However, the difficulty of assessing the curve itself has lend a way to using it's Average Precision Score. The higher the average precision score of a classifier, the better the classifier is at minimizing false positives. The average precision score of classifier h_i will denoted as $APS(h_i)$.

The average precision score of each classifier are:

$APS(h_1) = 0.649$
$APS(h_2) = 0.487$
$APS(h_3) = 0.517$
$APS(h_4) = 0.530$

$APS(h_5) = 0.684$

3.4 Bootstrap bounds on Average Precision Score

Since, the precision scores are unique according to each testing set. A method (Evans & Rosenthal, p. 355) devised to find the standard error by resampling from an empirical cumulative distribution function. Resamples are drawn with replacement on the testing data and then its average precision score is calculated. This process is repeated for 10^4 times in order to generate an empirical distribution function and which in turn will generate a standard error. Then a studentized γ-confidence interval of the average precision score can be computed as follows:

$$APS(h_i) \pm t_{(1+\gamma)/2}(n-1) \cdot \sqrt{Var(APS(h_i))} \text{ for } i = 1,2,3,4,5$$

Based on the bootstrap 95% and 99% confidence intervals (fig 3), the neural network has the tightest lower bound on the average precision score. While on the bootstrap simultaneous 95% confidence intervals (fig 3), there is a clear separation between the average precision score of the single layer neural network and the support vector machine with polynomial kernel. However, at the simultaneous 99% confidence interval all the classifier's average precision score overlaps again.

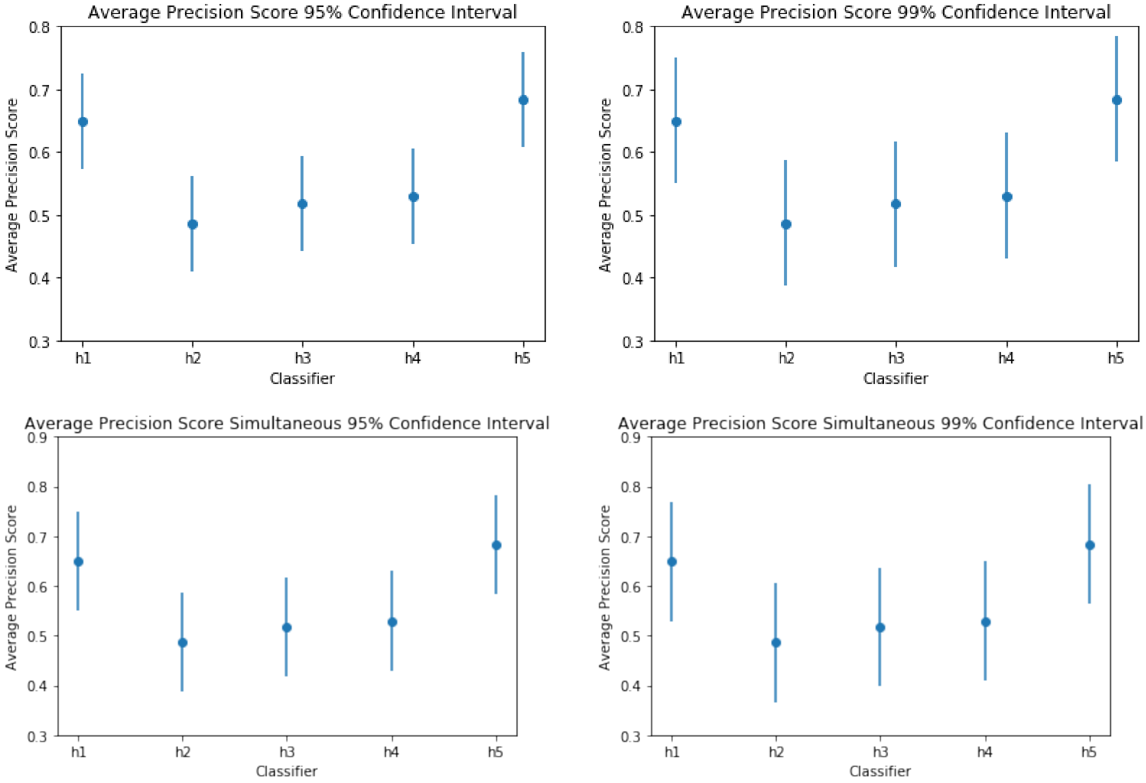

Figure 3. Bootstrap bounds for Average Precision Score

4 Improvements

The biggest omission of this experiment is hyper parameter tuning. Since, all the parameters were chosen by inspecting the training errors directly, it may have been that the final parameters chosen are not optimal. For future runs of this experiment this can be implemented. However, all results of this experiment can be replicated by using the same seed on the supplementary code.

5 Conclusion

Based on the zero one loss metric, it was discovered that there perhaps may not be a better classifier out of the five since the simultaneous confidence interval (fig 1) overlaps with each classifiers empirical risk. Also based on the average precision score, it was also discovered that at the bootstrap simultaneous 99% confidence interval (fig 3) there are no better classifier. However, based on the tightest upper bound of the empirical risk, the logistic regression will be the classifier of choice for this run of the experiment. While, based on the tightest lower bound of the average precision score, the single layer neural network will be the classifier of choice for this run of the experiment.

6 References

Evans, M. Rosenthal, J. S. Probability and statistics: the science of uncertainty. probability.ca (W.H. Freeman and Company, 2010).

Shalev-Shwartz, S. Ben-David, S. Understanding machine learning: from theory to algorithms. (Cambridge University Press, 2014).